IDEA based efficient task scheduling in Cloud Computing

Ravinder Kaur

M-Tech Student

ravinderkaur5688@gmail.com

Abstract Job scheduling is the process of allocating system resources to many different tasks by an operating system (OS). The system handles prioritized job queues that are awaiting CPU time and it should determine which job to be taken from which queue and the amount of time to be allocated for the job. Job Scheduling in Cloud Computing is an important task in this field to efficiently use resources in it. To address this thing in Fog computing 2 performance evaluation parameters are used those are the CPU execution time and the total amount of memory (allocated memory) needed by all tasks expected to be executed in the fog computing infrastructure. In the existing techniques of Job scheduling make span and response time were not upto mark. So in the proposed approach the IDEA based EDA is used for improvement in make span and response time. Because IDEA provide multiple solutions and then an optimal solution can be found for response time so that the jobs with less deadline span can be executing by maintaining the QoS. From the result section it is shown that the proposed approach is quite better than that of existing approach

I. BACKGROUND

Cloud Computing is one of the emerging technology to handle data, platform and infrastructure now a days as it offers a lot of flexibilities and features like storage, sharing and accessibility. In cloud computing various features affect the performance of its capabilities like scheduling of tasks, energy and speed etc[1]. Cloud computing has developed quickly and increased significant consideration since it gives adaptability and versatility to associations. Clouds in cloud computing is of a few kinds dependent on the versatility and pooling up of the assets. [1] Types are public, private, network, and hybrid clouds. Public clouds are accessible to the overall population in a compensation way and they are claimed by the cloud supplier. Private clouds are worked uniquely for a business or an association and they are constrained by that association or an outsider. In people group clouds, a few associations share the framework of the cloud to help certain network that has normal concerns. Hybrid clouds are blend of public, private, or network clouds [2][3].

II. TASK SCHEDULING

In cloud computing or general computing scheduling is a term by which planning is made to assign the jobs to various virtual machines. Virtual Machines are placed on hosts and execute the tasks that are assigned to them.

A scheduler in task scheduling may be said as heart of this whole process which is responsible for all the task assigning actions. This is used in a frequent manner as the jobs are to be distributed to Virtual Machines and assets are to be shared.[2] A scheduler do all these jobs according to certain planning that is called Task Scheduling Strategies. Task scheduling and resource management permit providers to maximize revenue and the utilization of resources up to their limits. In practice, in terms of the performance of cloud computing resources, the scheduling and allocation of resources are important hurdles. For this reason, researchers have been attracted to studies of task scheduling in cloud computing. Task scheduling is the process of arranging incoming requests (tasks) in a certain manner so that the available resources will be properly utilized.

III. RELATED STUDY

Pang, S.; et al. [1] defined an EDA-GA based technique in which results are firstly generated with the help of EDA that is Estimation of Distribution Algorithm and then the results of EDA are feeded to the Genetic Algorithm. Genetic Algorithm is a bio inspired algorithm in which the mutation and cross over rate processed the population generated through the EDA and then a solution using the above defined operations are generated and that is known as best optimal solution.

Saleh, H.; et al. [2] proposed and Improved Particle Swarm Optimization technique for allotment of all parameters in cloud architecture. The IPSO technique helps to adjust the memory heap size that is responsible for increased make and and lowering the QoS parameters. So IPSO based technique manages the heap size in RAM and then performed better in all aspects of Qos.

Sudheer, M.S.; et al. [3] proposed and algorithm that use Particle Swarm Optimization which helps the cloud to share bandwidth and memory in a decent way to the tasks that are to be scheduled. By fairly sharing the memory and bandwidth of the resources among the task scheduling the make span and other QoS parameters are improved to an extent.

Devarasetty, P.; et al. [4] introduced the assignment of system which designed using the information gathered from past experience that is break down and dependency upon QoS parameters. As per experience the algorithms are chosen on the basis of cost, time and stage of complexity. So an algorithm is put into execution which perform better results in terms of time and cost.

Djamegni, C.T.; et al. [5] proposed a scheduling scheme called Cost-Time Trade off Efficient Workflow Scheduling. This algorithm comprises of 4 steps as defined. Task Selection, Implicit Requested Instance Types Range (IRITR) evaluation, spare budget evaluation and VM selection. In this activation function extract the best suitable VMs Instance type in order to avoid over bidding and underbidding which may reduce the cost and improve the efficiency of VMs.

Kołodziej, E. et al. [6] develop a novel model based upon blockchain technique for cloud scheduling. This technique differs from other cotemporary techniques in the way that it tried to offload the implementation of the blockchain modules. Block Chain technique is used to provide the security for all the transactions and access of data in cloud. So this technique provide a secure and efficient make span based system for task scheduling in Cloud

Sen, P.; et al. [7] proposed a technique for scheduling in web administrations in multi cloud architecture. This technique is based upon how to get most effective results than that of choosing single cloud performance. In this technique the distributed cloud system is used for task scheduling with the cooperation among them which in turns reduce the load and make span.

Kaur, R. and Kaur, A. [8] proposed an improved assignment scheduling by checking out various task scheduling techniques in cloud computing. The proposed technique results in improve the QoS parameters of VMs. This technique confirms better results as it prevent the drawbacks of scheduling issues and techniques of task scheduling.

Zeng, J. et al. [9] proposed a technique that is based upon Q learning and also energy efficient in nature. It consists of 2 phases. In the first phase M/M/S queue is implemented using centralized task dispatcher. In the second phase the arriving tasks are assigned to each server by prioritize all the requests based upon task laxity and task life time. After this continuously-updating policy is used to assign the tasks to virtual machine.

Chong, C.K. et al. [10] proposed a method which aims to improve the performance of results by reducing the noisy data in initial population. To reduce the noisy data this paper used the Kalman filter and reduce the error rate and then improve the efficiency of this algorithm. By including the Kalman filter noisy data in population is reduced which in turns reduce the number of permutations and results in lower the execution time.

IV. METHODOLOGY

EDA-GA is Estimation of Distribution Algorithm using Genetic Algorithm which is a designed to check the multi objective task scheduling problem and it can be addressed by reducing the execution time of the tasks assigned. In this process following steps are to be used.[1]First of all the EDA is used to generate some feasible solutions with which the task scheduling can be done. After this Genetic Algorithm generates new solution based upon solution selected by the EDA that is EDA output is to be sent to GA to find out the best optimal solution[1]. Genetic Algorithm: Genetic Algorithm is the most emerged algorithm in Artificial Intelligence now days. This algorithm generated idea from Darwin's theory in which survival of the fittest allowed. This idea actually allocates the resources to the tasks on the basis of their fitness function. In this scheme tasks are not executed as a whole but according to the parameters of the tasks.
Improved Differential Evolution Algorithm which use the Kalman filter to reduce the noise in the population and then reduce the error rate in results and hence produce results in lower execution time and this way it overcome the drawbacks of Genetic Algorithm.[10] In the proposed EDA-IDEA scheme, first of all EDA algorithm will provide a set of solutions using Estimation Distribution. After this the set of solutions, that is provided by EDA, is send to IDEA which in turns provide the best optimal solution among all.[10] So using hybrid scheme of EDA and Improved Differential Algorithm problems of existing scheme may be overcome.

Fig 1 is the flow chart according to which the steps for research are executed. As shown in fig 4.1 whole research is divided into 4 parts.

First part is about the cloud Configuration.
In this part Host machine is configured that is the physical machine. Host machine is configured by providing RAM, SSD, Bandwidth and Processors.

After configuration of host machine, VMs are installed on each machine so that that jobs can be executed on VMs. VMs are configured by assign resources to them that are assigned from Physical or host machine. Total number of resources assigned to the VMs can-not exceed the total number of resources that a Physical Machine posses.

In the second part the existing scheme that is EDA-GA algorithm is executed on VMs. Jobs assigned to each VM will be executed using EDA-GA algorithm given below:

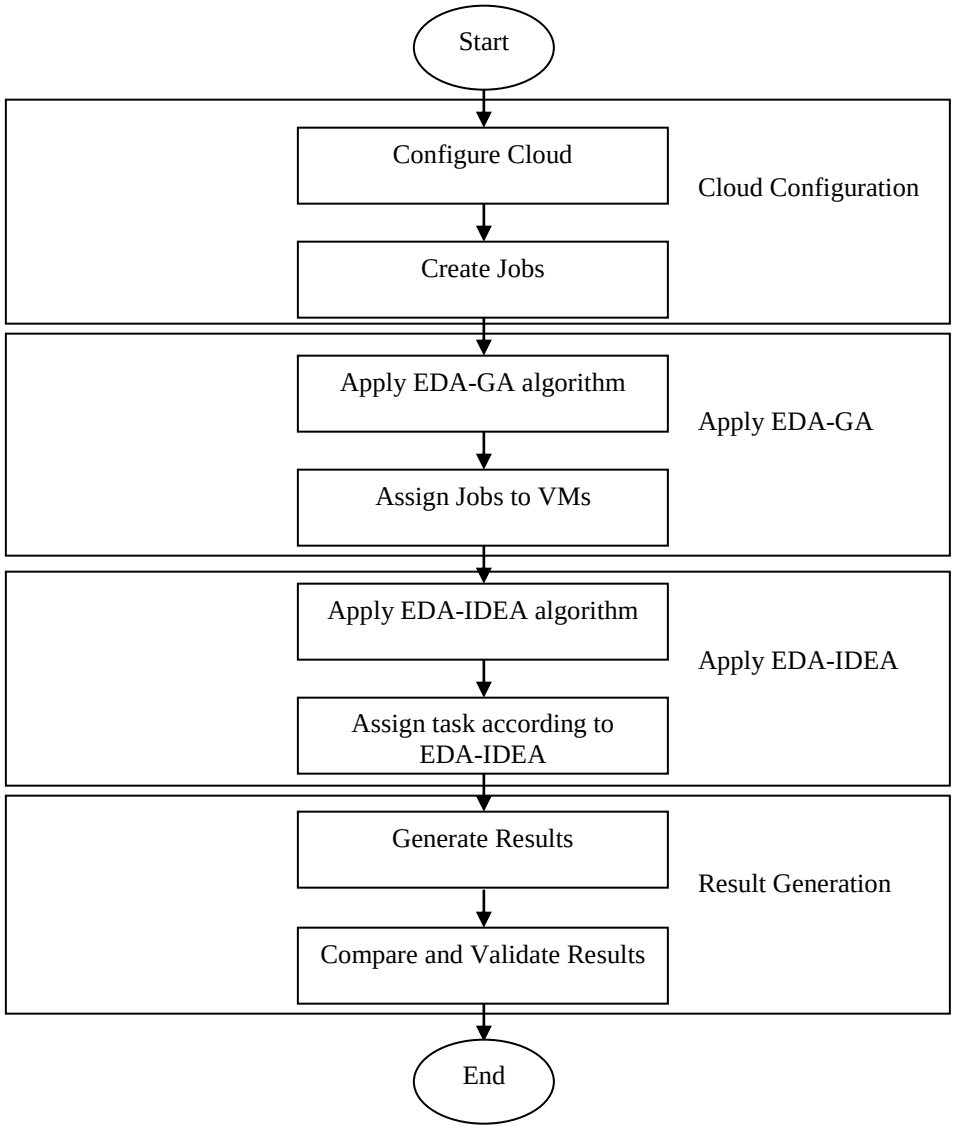

Fig 1: Flow Chart

Algorithm for EDA-GA
Input: Tasks (T), Virtual Machines (VMs)
Output: Optimal Solution for Scheduling of tasks
Initialize model of probability
while i <= i_{max} do
Sample Values of probability model for generation of population
Apply GA From Here
for every individual in population do
Calculate FVal
Sort individual according to desc order according to FVal
Select half excellent individual denoted as HEP,
for each particle in HEP do
apply crossover operations
apply mutation operation
Denote new population generated above as HNP
end for
for each particle in HEP and HNP do
Calculate FVal
Sort particles in descending order according to FVal
Extract top h% of the excellent particle form the excellent population generated above
end for
end for
GA Ends Here
Use excellent population for updation of probability model
i++

end while
Here FVal is the fitness function
GA has following drawbacks

- Because one cannot perform all the permutations, so one must bound to over fit the sample data.
- On the other hand if training is low then the accuracy and efficiency will also be low.

Due to the above drawbacks of GA, the proposed research will include IDEA optimization scheme that is defined below:
Algorithm: IDEA
Initialize population *P*
WHILE (l<max)
FOR (*j* = 0; *j* < *NP*; *j*++)
Select P[j_1], P[j_2], P[j_3]
Mutation
Choose Initial Candidate I[i] = P[j1] + F *(*P* [j2] - *P* [j3]).
Crossover
Choose final candidate *C*[*i*] by crossing over the genes of *P* [j] and I1[j] as follows:
FOR (*j1* = 0 ; *j1* < *NP* ; *j1*++)
IF (*U1*(0, 1) < *CR*)
I[*j*][*j1*] = *I*1[*j*][*j1*]
ELSE
I[*j*][*j1*] = *P*[*j*][*j1*]
END-FOR
Updating Population
I[*j*] = *inv*(*inv*(I[j]) + I)
*I= P*H1'*inv(H1*P*H1'+R)*
STEP 1.5: Evaluate *I*[*j*]
IF (*I*[*j*] is better than *P* [*j*])
P'[j] = I[*j*]
ELSE
P'[j] = *P* [j]
END-IF
END-FOR
P = *P*'
END-WHILE
END
K = Kalman gain value,
H = observation matrix,
H' = inverse of matrix *H*,
P = population
P' = population the next generation
I = candidate solution,
Using the above defined IDEA optimization algorithm the drawbacks of GA are overcome and GA is replaced with IDEA.

Simulation Parameters to be used are defined below
For Virtual Machines

Parameter	Value
Processors	1
SSD	10GB
RAM	512 MB
Bandwidth	512 Mbps
MIPS	1,00,000

For Host Machine

Parameter	Value
Processors	4
SSD	80GB
RAM	4 MB
Bandwidth	2 Mbps
MIPS	4,00,000

V. RESULTS AND DISCUSSION

Make Span: Make span can be defined as a ideal time for which a virtual machine is ideal when a task is executed and the new one is yet to assigned. Make span in a system should be low in cloud computing. If make span will be low then overall execution time of the system will be low, which means that system has executed the jobs in less time.
Response Time: Response time may be defined as an execution time that is possessed by a Virtual Machine to completely execute a task. Response Time of a system should be as low as possible. Low response time means that a system has executed a job in less time which in turns improve the overall performance of the system.
Over Utilization: Over Utilization of a VM may be defined as the jobs that are assigned to them when its own quota is executed. It is also defined as the migrated jobs that are assigned to a virtual machine.

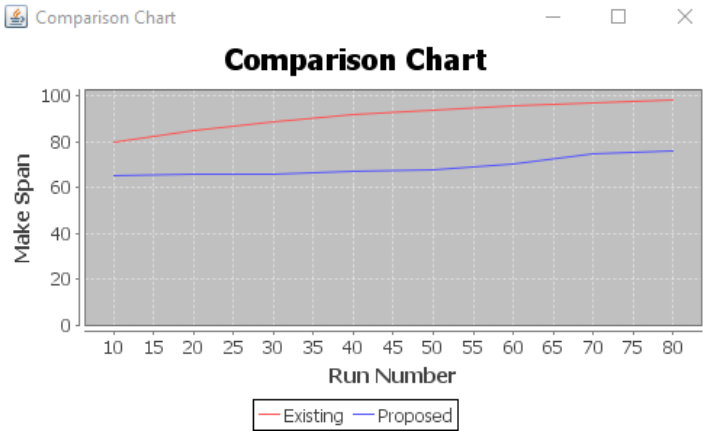

Fig 2: Cloudlet Size vs Make span

Fig 2 is the representation of comparative study for existing that is EDA-GA and proposed that is EDA-IDEA approach. Graph is showing the make span value against the cloudlet size. Make span can be defined as a ideal time for which a virtual machine is ideal when a task is executed and the new one is yet to assigned. Fig 5.1 is showing the make span of EDA-IDEA based scheduling scheme as compare to EDA-GA. In EDA-GA scheduling the make span is approx 98 ms where in case of EDA-IDEA it is below 80ms.

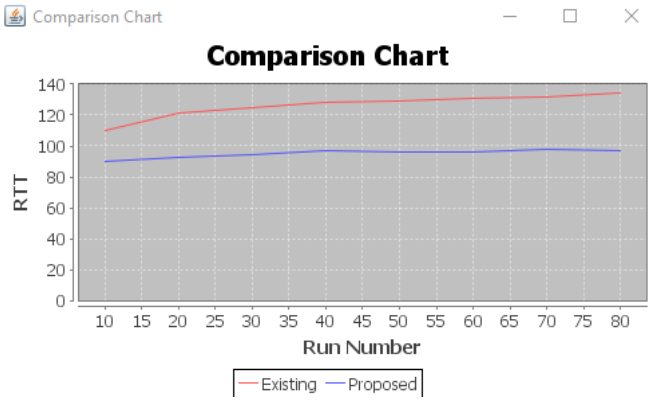

Fig 3: Response Time vs Jobs

Fig 3 is the representation of comparative study for existing that is EDA-GA and proposed that is EDA-IDEA approach. Graph is showing the response time value against the number of Jobs assigned. Make span can be defined as a ideal time for which a virtual machine is ideal when a task is executed and the new one is yet to assigned. Fig 5.3 is showing the response time of EDA-IDEA based scheduling scheme as compare to EDA-GA. In EDA-GA scheduling the response time is approx 137 ms where in case of EDA-IDEA it is below 100ms.

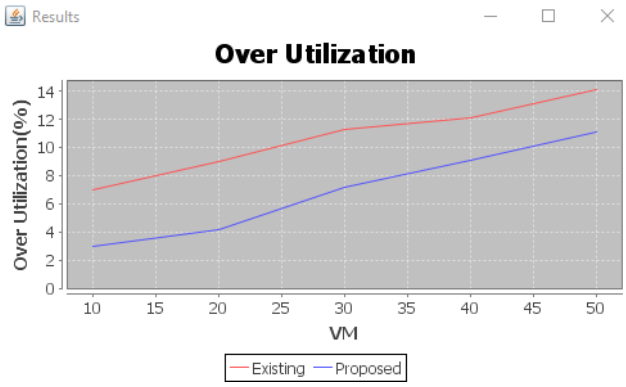

Fig 4: Over Utilization

Fig 4 is the representation of comparative study for existing that is EDA-GA and proposed that is EDA-IDEA approach. Graph is showing the Over utilization value against the number of Jobs assigned. Over Utilization of a VM may be defined as the jobs that are assigned to them when its own quota is executed. It is also defined as the migrated jobs that are assigned to a virtual machine. Fig 5.5 is showing the over utilization of EDA-IDEA based scheduling scheme as compare to EDA-GA. In EDA-GA scheduling the over utilization is approx 14% where in case of EDA-IDEA it is below 12%.

Table 1: Comparison of existing and proposed scheduling algorithm

Technique / Parameters	Existing	Proposed
Make span	98	79
Over Utilization	14	11
Response Time(RTT)	130	100

In Table 1 a comparison of existing and proposed scheduling is shown. As makespan in existing technique is 98ms where as in proposed it is 79. Overutilization in case of existing is 14 where as in proposed it is 11. Response time is 130 in case of existing and 100 in case of proposed technique.

VI. CONCLUSION

This execution points towards the foundation of execution subjective examination on make range in VM task allotment and procedure as per their cutoff time, at that point actualized in CloudSim with Java language. Here significant pressure is given on the investigation of dead line based errand planning calculation with heterogeneous assets of the cloud, trailed by near review of different calculations in cloud computing as for adaptability, homogeneity or heterogeneity and procedure booking.

A past report additionally demonstrates change of MIPS will influence the reaction time and increment in MIPS versus VM diminishes the reaction time. At the point when picture size of VM is actualized against the VM data transmission then no huge impact is found on reaction time and it stays consistent for which these boundaries are examined. In any case, if there should arise an occurrence of Cloudlet long length versus Host transfer speed an example is seen in which reaction time increments in proportionate way.

Utilizing the IDEA based EDA methodology the decrease in the response time and make span of the different procedures are accomplished as appeared in results. From the outcomes unmistakably the proposed framework utilized the undertaking cutoff time as info boundary to improve results.

Future Scope:

1. Deadline based task scheduling of VMs allows a server administrator to move a running virtual machine or application among different physical machines without disconnecting the client or application.
2. Total allocation time and downtime are two key performance metrics that the clients of a VM service care about the most, because they are concerned about service degradation and the duration that the service is completely unavailable.
3. Cloud computing is a pivotal factor and here as a future course of work similar other parameters can be investigated against each other to find the effect on load balancing.

VII. REFERENCES

1. Pang, S.; Li, W.; He, H.; Shan, Z.; Wang, A.X.; "An EDA-GA Hybrid Algorithm for Multi-Objective Task Scheduling in Cloud Computing", IEEE, vol: 7, 2019, pp: 146379-146389
2. Saleh, H.; Nashaat, H.; Saber, W.; Harb, H.M.; "IPSO Task Scheduling Algorithm for Large Scale Data in Cloud Computing Environment", IEEE, vol: 7, 2019, pp: 5412-5420
3. Sudheer, M.S.; Krishna, M.V.; "Dynamic PSO for Task Scheduling Optimization in Cloud Computing", International Journal of Recent Technology and Engineering, vol: 8, 2019, pp: 332-338
4. Devarasetty, P.; Reddy, S.; "Research of Task Management and Resource Allocation in Cloud Computing", International Journal of Innovative Technology and Exploring Engineering, vol: 8, 2019, pp: 938-941
5. Djamegni, C.T.; Kamla, V.C..; Mboula, J.E.N.; "Cost-time trade-off efficient workflow scheduling in cloud", Simulation Modelling Practice and Theory, 2020, Vol: 103, pp:102-107
6. Kołodziej, E.; Wilczynski, A.; "Modelling and simulation of security-aware task scheduling in cloud computing based on Blockchain technology", Simulation Modelling Practice and Theory, vol: 99, 2020
7. Sen, P.; Sarddar, D.; Sinha, S.K.; Pandit, R.; "Web Service Scheduling in Multi-Cloud Environment", International Journal of Computer Sciences and Engineering, vol: 7, 2019, pp: 30-38
8. Kaur, R.; Kaur, A.; "Efficient job scheduling algorithm based on weighted average priority in cloud computing", International Journal of Advance Re-

search, Ideas and Innovations in Technology, vol: 5, 2019, pp: 354-349
9. Zeng, J.; Kang, K.; Zhao, Y.; Fan, X.; Ding, D.; "Q-learning based dynamic task scheduling for energy-efficient cloud computing", Future Generation Computer Systems, vol: 108, 2020, pp: 361-371
10. Chong, C.K.; Mohamad, M.S.; Deris, S.; Shamsir, M.D.; Choon, Y.W.; Chai, L.E.; "Improved Differential Evolution Algorithm for Parameter Estimation to Improve the Production of Biochemical Pathway", International Journal of Artificial Intelligence and Interactive Multimedia, Vol: 1, No 5, pp: 22-29
11. Zang, L.; Zhou, L.; Salah, A.; "Efficient scientific workflow scheduling for deadline-constrained parallel tasks in cloud computing environments", Information Sciences, vol: 531, 2020, pp: 31-46
12. Sanaj, M.S.; Prathap, P.M.J.; "Nature inspired chaotic squirrel search algorithm (CSSA) for multi objective task scheduling in an IAAS cloud computing atmosphere", Engineering Science and Technology, an International Journal, 2019
13. Refieyan, E.; Khorsand, R.; Ramezanpour, M.; "An adaptive scheduling approach based on integrated best-worst and VIKOR for cloud computing", Computers & Industrial Engineering, Vol: 140, 2020
14. Chaudhary, D.; Kumar, B.; "Cost optimized Hybrid Genetic-Gravitational Search Algorithm for load scheduling in Cloud Computing", Applied Soft Computing, Vol. 83, 2019
15. Aloboud, E.; Kurdi, H.; "Cuckoo-inspired Job Scheduling Algorithm for Cloud Computing", Procedia Computer Science, vol: 151, 2019
16. Sharma, N.; Tyagi, S.; "A Comparative Analysis of Min-Min and Max-Min Algorithms based on the Makespan Parameter", International Journal of Advanced Research in Computer Science, Vol: 8, No: 3, 2017, pp: 1038-1041
17. Miglani, N.; Sharma, G.; "Modified Particle Swarm Optimization based upon Task categorization in Cloud Environment", International Journal of Engineering and Advanced Technology, Vol: 8, Issue: 4C, 2019, pp: 67-72
18. Patel, S.; Bhoi, U.; "Priority Based Job Scheduling Techniques In Cloud Computing: A Systematic Review", International Journal Of Scientific & Technology Research, Vol: 2, Issue: 11, 2013, pp: 147-152
19. Moon, Y.J.; Yu, H.; Gil, J.M.; Lim, J.B.; "A slave ants based ant colony optimization algorithm for task scheduling in cloud computing environments", Human-centric Computing and Information Sciences, Vol: 7, number: 1, 2017
20. Srichandan, S.; Kumar, T.A.; Bibhudatta, S.; "Task scheduling for cloud computing using multi-objective hybrid bacteriaforaging algorithm", Future Computing and Informatics Journal, vol: 3, 2018, pp: 210-230